Global Journal of Engineering and Technology Advances

eISSN: 2582-5003

Cross Ref DOI: 10.30574/gjeta

Journal homepage: https://gjeta.com/

(RESEARCH ARTICLE)

Check for updates

Development of a terrorism threat detection system using natural language processing

Adesoji Adedeji Adegbola *, Oluwadamilare Enoch Adewumi, Temple Ajimaba and John Toluwani Osazuwa

Software Engineering, School of Computing, Babcock University, Nigeria.

Global Journal of Engineering and Technology Advances, 2026, 27(02), 007-018

Publication history: Received on 17 March 2026; revised on 04 May 2026; accepted on 06 May 2026

Article DOI: https://doi.org/10.30574/gjeta.2026.27.2.0107

Abstract

Terrorism remains a major global security challenge, and the increasing use of digital communication platforms has made it easier for individuals or groups to spread threatening content and coordinate harmful activities. As a result, there is a growing need for intelligent systems capable of automatically detecting potential threats from large volumes of textual data. This study focuses on the development of a terrorism threat detection system that utilizes Natural Language Processing (NLP) and machine learning techniques to analyze text and identify messages that may contain terrorism-related threats.

The system was developed using a dataset consisting of various text samples categorized as terrorism-related threats, suspicious messages, and normal or non-threatening communication. The dataset underwent several preprocessing steps including text cleaning, tokenization, stop-word removal, and stemming or lemmatization. The processed text was then converted into numerical form using vectorization techniques enabling it to be used by machine learning models. Suitable classification models were trained on a training dataset, while a separate testing dataset was used to evaluate performance.

The evaluation of the system was carried out using performance metrics such as accuracy, precision, recall, F1 score, and Matthews Correlation Coefficient (MCC). The results demonstrated that the model effectively identified patterns associated with terrorism-related threats in textual data, achieving perfect scores across all metrics over three training epochs.

The system showed promising performance in distinguishing between threatening and non-threatening messages, indicating that machine learning and NLP techniques can be effectively applied to support automated threat detection and enhance security monitoring systems.

Keywords: Terrorism Threat Detection; Natural Language Processing; Threat Classification; BERT; Web-based System

1. Introduction

Over the years, Natural Language Processing (NLP) has developed into a powerful tool in the field of security. Recent studies demonstrate that NLP-powered models can achieve detection accuracies exceeding 90% when identifying extremist and terrorist-related content on online platforms [2]. This underscores its transformative role in cybersecurity, particularly through the use of deep learning techniques and contextual models that enable the efficient processing and analysis of vast amounts of unstructured data.

* Corresponding author: Adesoji Adedeji Adegbola.

Historically, terrorism threat detection relied heavily on manual intelligence gathering, human informants, and conventional surveillance methods. Although effective to some extent, these approaches were limited by human analytical capacity, delayed response times, and subjective interpretation of data. As online communication continues to evolve in complexity and scale, the need for automated, intelligent detection systems has become critical.

Advancements in NLP and deep learning have introduced powerful tools capable of analyzing large volumes of unstructured text in real time. Transformer-based models such as BERT have demonstrated strong contextual understanding and improved accuracy in classification tasks. These models are capable of capturing semantic meaning, contextual relationships, and subtle linguistic patterns, making them suitable for detecting coded or indirect threat language.

Despite these advancements, many existing systems still struggle with contextual nuance, sociocultural language variations, and maintaining consistent performance across diverse datasets. This highlights the need for adaptive, scalable, and context-aware systems capable of robust threat analysis.

This paper outlines the design, development, and implementation of a terrorism threat detection system. It explores the system architecture, key features, and technologies used, while also addressing the limitations associated with its development. The goal is to provide a scalable, context-aware terrorism threat detection system that can meet the evolving needs of terrorism prevention.

1.1. Statement of the Problem

While existing NLP and machine learning models demonstrate potential in terrorism detection, they often fail to capture nuanced contextual meanings, adapt to sociocultural variations in language use, and maintain consistent accuracy across heterogeneous data sources. These shortcomings delay the timely detection and prevention of terrorist activities, emphasizing the necessity for scalable and culturally aware AI-driven solutions that can reliably process both textual and multimedia content. To overcome these limitations, this study proposes the use of the BERT model and other NLP techniques to develop adaptive, context-aware models capable of analyzing unstructured data in real time.

1.2. Aim and Objectives

The aim of this project is to develop a system that will accurately identify and classify terrorism-related threats and perpetrators' aims from unstructured data. The objectives to achieve this aim include:

- To adopt a transformer-based NLP model capable of extracting terrorism-related entities and indicators from unstructured text across diverse sources.
- To design a terrorism threat detection system that integrates the selected model.
- To implement the system using advanced machine learning techniques, integrating them on varying types of inputs within datasets.
- To evaluate the system's performance through comprehensive metrics with particular emphasis on accuracy to ensure timely and reliable threat detection.

2. Literature review

The evolving landscape of global security, characterized by the rapid dissemination of information and the dynamic nature of threats, necessitates advanced threat detection systems. Traditional security systems often suffer from significant limitations, including delays in information processing, inefficiencies in analysis, and a general lack of real-time awareness, making them inadequate for addressing modern security challenges. These conventional approaches frequently produce generic alerts, lack personalization in threat assessment, and facilitate slow communication across relevant agencies. The underlying data sources and usages in terrorism and radicalization research are constantly evolving, as terrorist groups increasingly leverage social media to propagate content rather than structured blogs or forums, rendering detection processes quickly outdated [3]. Consequently, there is a pressing need for technological interventions, particularly artificial intelligence (AI), real-time data analytics, and predictive modeling, to enhance the ability to identify and respond to emergent threats.

In response to these challenges, contemporary research and development efforts have explored various technological solutions for threat detection. Innovations include the implementation of multi-channel alert systems utilizing SMS, mobile applications, and social media monitoring to capture a broader spectrum of information [4]. The integration of the Internet of Things (IoT) with machine learning (ML) models has been identified as a critical component for

processing large-scale data and generating actionable insights [5]. GPS/GSM tracking and automated dispatch systems have also been developed to improve response coordination. In terms of system architecture, multi-tier designs, cloud computing, and edge computing are increasingly employed to handle the volume and velocity of data generated from diverse sources. Performance metrics such as latency, accuracy, and success rates are crucial for evaluating the efficacy of these systems, with many studies reporting improved classification accuracies using advanced NLP and ML techniques. In particular, deep neural networks, especially models like BERT, have made significant strides in extracting contextual and semantic features from text, thereby improving classification accuracy compared to older methods like TF-IDF [6].

Despite these advancements, significant research gaps persist in the domain of terrorism threat detection. Many existing infrastructures are outdated and struggle to adapt to the rapidly evolving nature of extremist discourse. There is a notable lack of modular, scalable systems that can seamlessly integrate new data sources and analytical techniques [3]. Furthermore, weak inter-agency coordination and a lack of resilience against network failures or cyberattacks remain critical vulnerabilities. In specific contexts such as Nigeria and other developing regions, challenges are exacerbated by poor signal coverage and protocol incompatibility. Neglected areas in current research include the limited incorporation of citizen participation in threat reporting and the development of adaptive messaging systems that can tailor communications based on dynamic threat assessments. Reviews on NLP approaches for extremism often lack depth in describing diverse techniques, focusing mainly on detection rather than the broader data mining process [1]. Addressing these gaps necessitates the design and implementation of a robust terrorism threat detection system that leverages cutting-edge NLP techniques to provide real-time, context-aware, and adaptable intelligence.

3. Methodology

To achieve the stated objectives, the research followed four sequential phases: model selection, model integration (system design), implementation, and evaluation.

3.1. Model Selection

The initial phase focused on selecting a transformer-based NLP model capable of extracting terrorism-related entities and indicators from unstructured text. Following a comprehensive literature review, Bidirectional Encoder Representations from Transformers (BERT) was selected due to its proven accuracy and robustness in classification tasks. Studies such as Abdalsalam et al. [6] show that BERT effectively captures complex semantic relationships and nuanced language patterns, making it well-suited for detecting terrorism-related content.

The model was implemented using the Hugging Face Transformers library on a PyTorch backend, enabling efficient development and fine-tuning. Evaluation was conducted using the Global Terrorism Database (GTD), manually generated synthetic examples, and datasets from Kaggle, which provide structured and semi-structured terrorism-related data for testing preprocessing and model performance. The evaluation focused on preprocessing effectiveness, entity extraction accuracy, and hardware resource consumption. Based on these results, the final system architecture adopted a stack comprising BERT, Hugging Face Transformers, and PyTorch, balancing accuracy, scalability, and efficiency.

3.2. System Design and Model Integration

The second phase focused on designing the terrorism threat detection system by integrating the selected NLP model to enable accurate threat detection. This phase involved collecting stakeholder requirements through surveys and classifying them into functional and non-functional requirements, which guided the overall system design and ensured alignment with user needs.

Several modelling techniques were employed to refine these requirements into a feasible system architecture. A conceptual model was first developed to provide a high-level representation of system components and their relationships. UML use case and sequence diagrams were then used to illustrate interactions between users and system components, as well as internal process flows. Additionally, a pipe-and-filter architecture model was adopted to represent the structured flow of data from input collection through processing stages to output generation. The functional decomposition diagram further outlined the system workflow, detailing how raw inputs are transformed into actionable threat reports, while an entity-relationship diagram was used to define how system entities interact in handling and analysing text and audio data. From an implementation perspective, the system utilized a React frontend with TypeScript to provide an interactive user interface, and a multimodal API was integrated to support audio-to-text conversion.

3.3. Implementation

The third phase involved implementing the system based on the developed design. This included data collection and preparation, model training, and the configuration of required libraries for tokenization and sentiment analysis. A Python-based backend was configured with the selected model stack to form a unified NLP data pipeline, enabling seamless processing from input to output. To extend system capability to audio inputs, Google Gemini's API was configured for Automatic Speech Recognition (ASR), enabling speech-to-text conversion and allowing the system to process both textual and spoken data. The core threat detection component was developed and tightly integrated with Named Entity Recognition (NER) to identify, extract, and classify key information, including intent and other relevant entities, into predefined categories. Domain-specific datasets were utilized to improve contextual understanding and ensure accurate classification of terrorism-related content.

3.4. Evaluation

The fourth phase focused on evaluating the system's performance within a simulated evaluation environment. By utilizing sample inputs, the system was tested iteratively to observe performance metrics and record faults, allowing for continuous refinement. To achieve a comprehensive assessment, several key metrics were employed. Accuracy was measured as the percentage of correctly classified messages, while Precision determined how many flagged threats were genuinely threats. Recall was used to measure how many actual threats the system successfully caught, and the F1 Score provided a balance between Precision and Recall. Additionally, the Matthews Correlation Coefficient (MCC) served as a single-figure reliability score that accounted for all four classification outcomes — True Positives, True Negatives, False Positives, and False Negatives. By analyzing these measures alongside the false alarm and missed detection rates, the system was finely tuned for maximum dependability.

4. System design

4.1. Software Architectural Design

The terrorism threat detection system utilizes a pipe-and-filter architecture, selected for its modularity and sequential efficiency. This design allows the system to be divided into independent processing stages, ensuring that data flows smoothly through organized transformations. By decoupling these functions, the architecture provides a scalable and maintainable framework where components can be updated or improved without disrupting the entire pipeline.

The process begins with the ingestion of raw textual and audio data from various sources, such as social media, intercepted communications, and live feeds. To ensure consistency, the audio stream is immediately routed through a speech-to-text conversion stage, which unifies all inputs into a single textual format. Once unified, the data enters a pre-processing stage where it is cleaned, standardized, and tokenized to remove irrelevant noise. This refined text is then forwarded to Named Entity Recognition (NER), which extracts key identities such as intent, weapons, locations, and organizations. The subsequent NLP analysis component evaluates the deeper context, sentiment, and linguistic patterns of the text to detect subtle indicators of extremism. Based on these insights, the scoring and classification stage assigns a threat level and categorizes the content as threatening or non-threatening. Finally, the report generation component compiles these findings into structured summaries and visual reports, enabling security personnel to interpret results quickly and take proactive action.

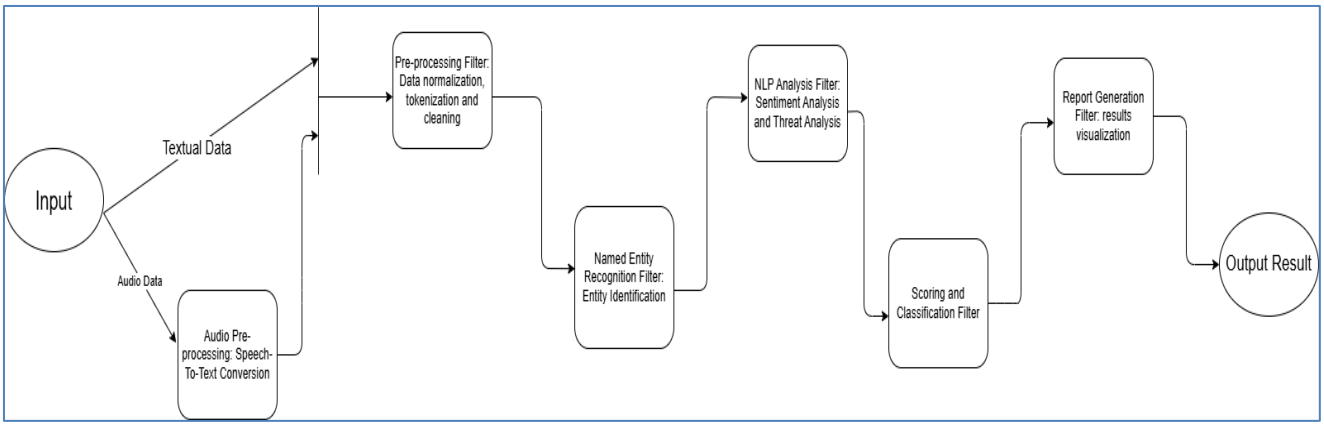

Figure 1 Software Architectural Design for the Terrorism Threat Detection System

4.2. Conceptual Design

The system receives two primary input types — textual data (social media, emails, forums) and audio data (recorded speeches, intercepted calls). Since NLP techniques work with text, audio is first converted to written text via speech recognition, creating a single unified stream for processing. This text then enters a pre-processing stage where it is cleaned, tokenized, and stripped of noise such as special characters or redundant content. Next, NER identifies key entities including weapons, locations, and organizations, helping the system understand the subjects and context of each communication. The processed text is then passed through the core NLP analysis stage, which combines sentiment analysis to detect hostile or aggressive language with threat detection mechanisms that identify extremist patterns and contextual cues, going beyond simple keyword matching. Finally, a scoring and triage module evaluates severity by assigning a threat score based on language intensity, identified entities, and contextual indicators, and the results are compiled into a structured output report for analysts.

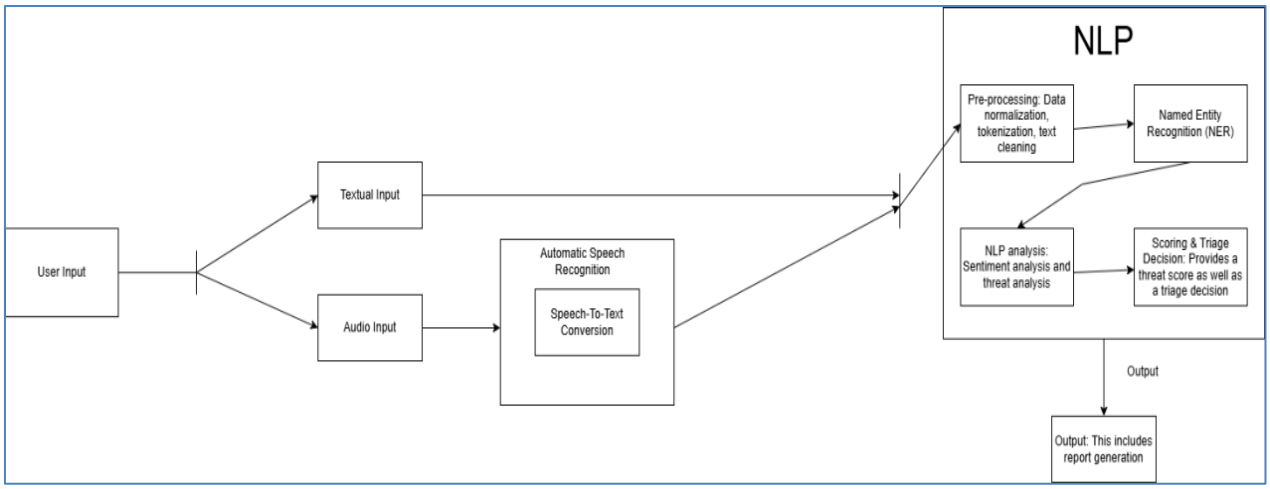

Figure 2 Conceptual Diagram for the Terrorism Threat Detection System

4.3. Functional Decomposition

The functional decomposition diagram illustrates how the system is divided into its core functional components and sub-processes. The system integrates NLP techniques including Named Entity Recognition, Sentiment Analysis, and Semantic Analysis to detect and assess potential terrorism-related content through the following five functional blocks:

- Input Data Processing: The system collects incoming textual or audio data, validates it, routes it by type, and applies preprocessing including tokenization, lowercasing, and stop-word removal. Audio inputs are converted to text via speech-to-text conversion, ensuring all inputs are standardized for uniform NLP-based analysis.
- Named Entity Recognition (NER): The system uses NLP techniques to extract named entities such as intentions, weapons, and other terrorism-related terms, isolating relevant threat-related elements as the foundation for further semantic and sentiment analysis.
- NLP Analysis: This stage interprets the text's meaning and emotional tone through sentiment analysis, threat detection (identifying patterns or language that could indicate terrorist intentions), and semantic analysis to understand context and entity relationships beyond keywords.
- Threat Scoring: The system assigns numerical values to parameters such as sentiment intensity and frequency of threat-related entities, then aggregates these scores to classify the threat as low, medium, or high severity, enabling prioritization for human review or automated alerting.
- Report Generation: The final stage produces a summarized threat report containing identified entities, sentiment scores, and the overall threat classification, presenting the system's analysis in a clear, structured format for security analysts and decision-makers.

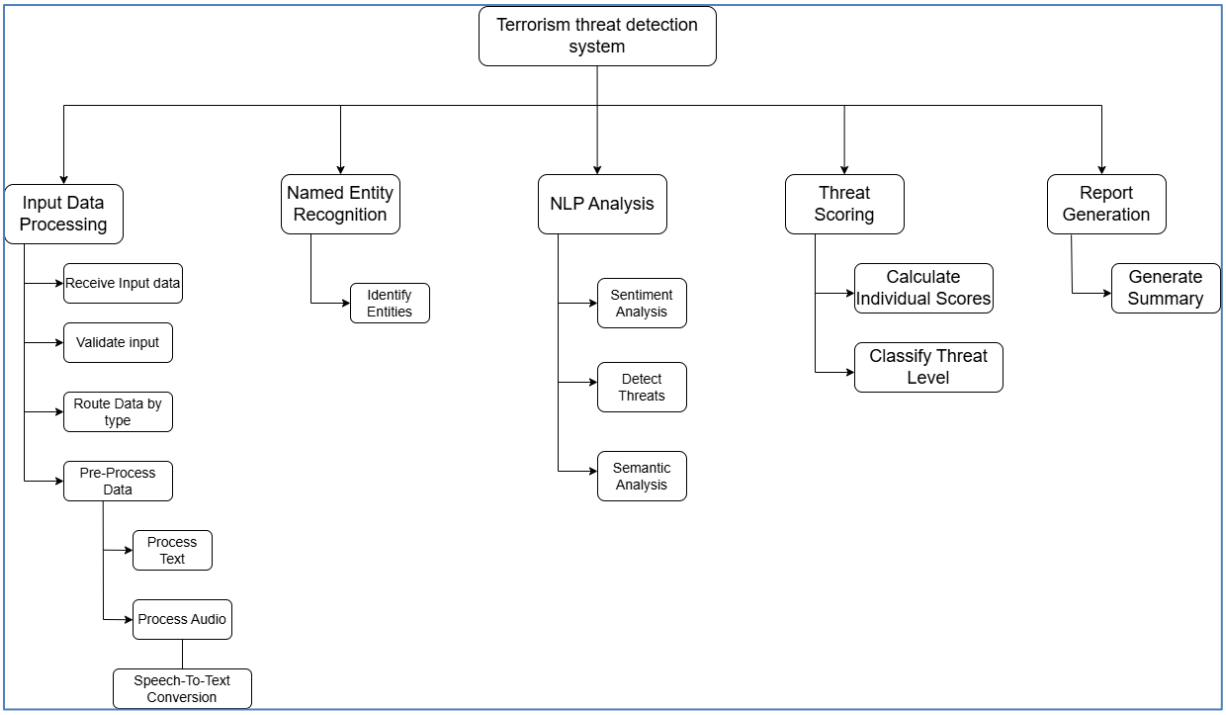

Figure 3 Functional Decomposition Diagram

5. Implementation tools and technologies

The terrorism threat detection system was implemented using an architecture that integrates a web-based monitoring interface with a data processing backend designed for intelligence analysis. The primary interface for analysts was developed as a web application using React, TypeScript, and Tailwind CSS, enabling an interactive and responsive environment for monitoring and managing detected threats. This dashboard provides users with access to processed text, identified entities, detected threat levels, and associated contextual information. Analysts can view detailed reports and monitor system outputs in real time, with complex analytical results presented in a structured and user-friendly format.

The backend infrastructure is powered by Supabase, which provides a scalable and secure environment for data storage, authentication, and real-time communication. Built on a PostgreSQL database, the backend handles the storage of raw inputs, processed text, extracted entities, and classification results. It also manages user authentication, ensuring that only authorized personnel can access sensitive data. Real-time subscriptions enable continuous updates between the backend and the web interface, ensuring that newly detected threats and analysis results are immediately visible to analysts.

To support multimodal data processing, the system accepts both textual and audio inputs. Textual data may originate from social media platforms, online communications, or digital reports, while audio data may include recorded conversations or intercepted voice communications. Audio inputs are processed through Google Gemini's ASR API, which converts spoken language into text for analysis alongside other textual inputs. Due to current constraints, transcription is limited to English, but the architecture allows for future expansion to support multiple languages.

Once all inputs are in textual form, they are processed using NLP techniques including the BERT model in conjunction with rule-based classification techniques. This component performs sentiment evaluation, contextual understanding, and threat classification, examining content for linguistic patterns, keywords, and contextual indicators associated with threatening intent. The overall architecture enables seamless data flow from input acquisition to final analysis and visualization, supporting continuous monitoring, automated threat detection, and efficient intelligence analysis.

6. Results

6.1. Model Evaluation

To measure the effectiveness of the terrorism threat detection system, several evaluation metrics were used. Accuracy measures the percentage of correctly classified messages, Precision measures how many predicted threats were actually true threats, and Recall measures how many actual threats were correctly detected. The F1 score was used to balance Precision and Recall and is calculated as:

$$F1 = 2 \times (Precision \times Recall) / (Precision + Recall)$$

Additionally, the Matthews Correlation Coefficient (MCC) was included as a more reliable single-figure measure of classification quality, as it takes into account all four classification outcomes — true positives, true negatives, false positives, and false negatives — and is particularly well-suited for balanced datasets.

The model was trained over three iterations (epochs), with performance evaluated at the end of each. Validation loss decreased consistently, indicating steady convergence without overfitting. As shown in Table 1, all five metrics achieved a perfect score of 1.0 across all epochs, demonstrating that the model classified threat-related content with exceptional reliability. This outcome can be attributed to the application of a 0.3 dropout rate as an effective regularization mechanism, and to BERT's pre-trained contextual language representations, which provided a strong foundation requiring relatively little fine-tuning to distinguish between threatening and non-threatening language.

Table 1 Model Validation Metrics Across Training Epochs

Epoch (Iteration)	Validation Loss	Accuracy Score	Precision Score	Recall	F1 Score	MCC
1	0.000006	1.0000	1.0000	1.0000	1.0000	1.0000
2	0.000002	1.0000	1.0000	1.0000	1.0000	1.0000
3	0.000001	1.0000	1.0000	1.0000	1.0000	1.0000

While the system performed exceptionally well on the structured and balanced dataset used in this study, it is important to acknowledge that perfect evaluation scores do not necessarily reflect real-world performance. In practice, terrorism-related threats are often expressed through coded language, slang, sarcasm, or culturally specific references that may not be well represented in the training data. As such, the model would benefit significantly from evaluation on a larger, noisier, and more linguistically diverse dataset in future work.

6.2. System Evaluation

Mean Opinion Score (MOS) was collected from four users, each evaluating the Terrorism Threat Detection system on two performance dimensions using the ITU-T P.800 five-point scale, where 1 represents the poorest performance and 5 represents the best. Table 2 presents the results per metric.

Table 2 TTD Mean Opinion Score (MOS) per Metric

Metric	MOS (Mean ± SD)
Detection accuracy	4.25 ± 0.50
Response speed	4.00 ± 0.82

Detection accuracy received the highest user ratings, with a MOS of 4.25 ± 0.50. This outcome suggests that users consistently regarded the system as reliable in identifying threats from their inputs. The low standard deviation of 0.50 indicates strong agreement among evaluators, thereby increasing confidence in this result. Response speed achieved a MOS of 4.00 ± 0.82, indicating generally rapid system performance. The higher standard deviation, however, suggests greater variability in user experience, which may be attributed to differences in input complexity. Both metrics fall within the "Good" performance band (4.0–5.0), suggesting that the system performs adequately across its core

functions. However, the elevated standard deviation observed in response speed warrants further investigation, particularly with respect to how input length and complexity influence detection latency.

6.3. System Interface

The implemented system resulted in a fully functional web-based dashboard for security analysts, capable of real-time analysis of both textual and audio inputs. The sign-up and sign-in pages serve as the secure access gateway, featuring a dark-themed interface with neon teal accents that emphasizes data security and system professionalism. Once authenticated, users can access the Threat Analyzer dashboard, which serves as the core functional hub of the system.

The Threat Analyzer provides a sophisticated interface for multi-format data ingestion, supporting direct text entry, document uploads, and audio uploads. The system's BERT-based model performs deep linguistic analysis beyond simple keyword matching, generating real-time results that categorize detected threats into specific risk levels such as "Direct Intent" or "Escalation Language." The Scan History page provides a centralized record of all previous threat detection activities, displaying a detailed log of analyzed data including the original input text, the source format, and the corresponding analysis results. Each entry in the history log preserves the critical metadata from the BERT-based model, showing the specific threat tags identified and the final confidence percentage, enabling analysts to track recurring patterns over time and maintain an audit trail for further investigation and reporting.

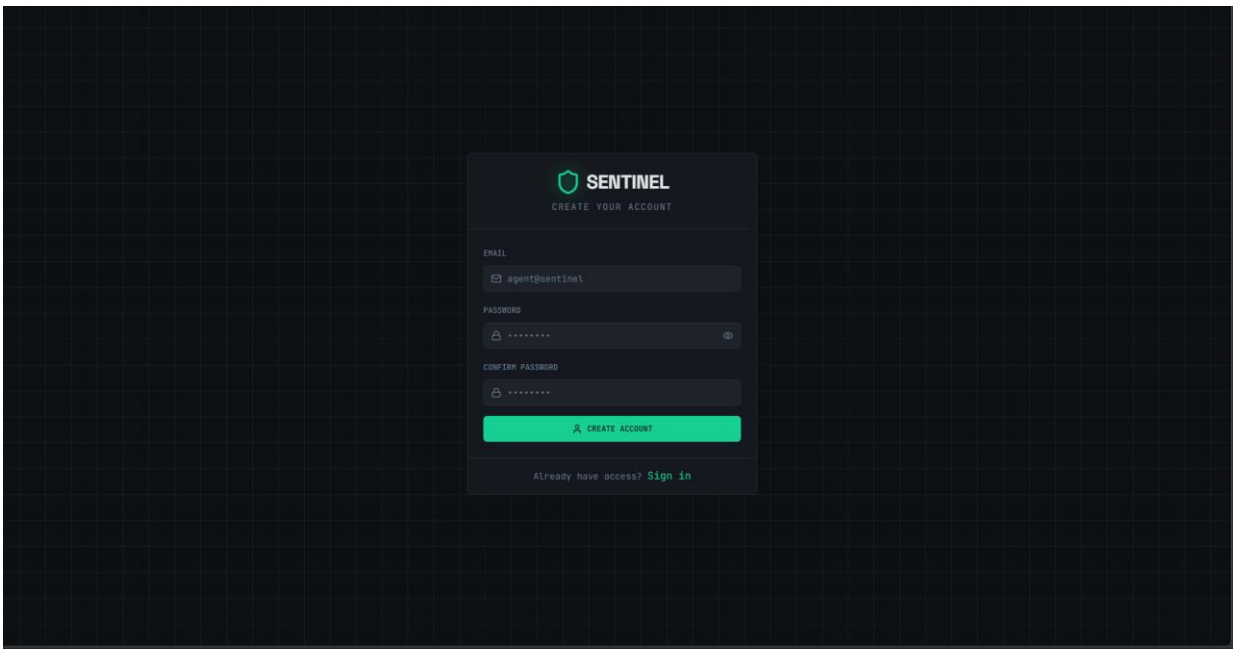

Figure 4 Sign Up Page

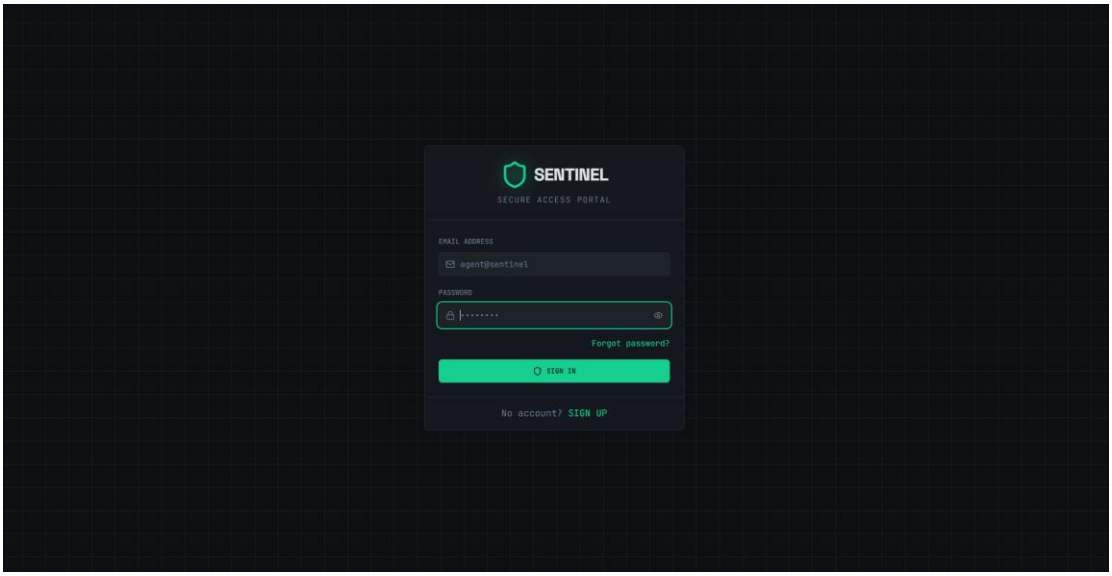

Figure 5 Sign In Page

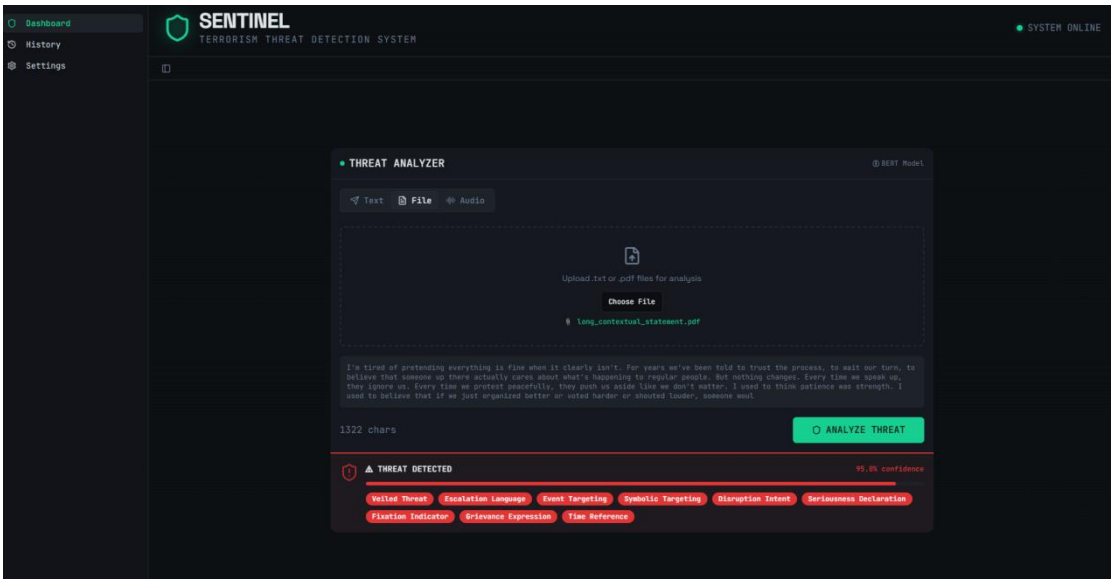

Figure 6 Threat Analyzer (File Format)

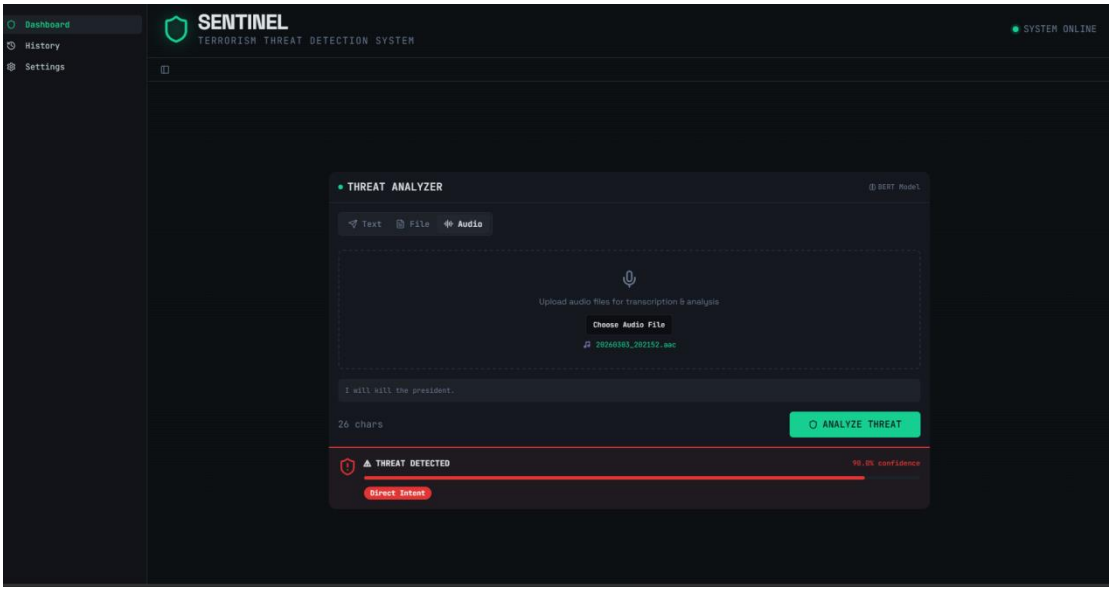

Figure 7 Threat Analyzer (Audio Format)

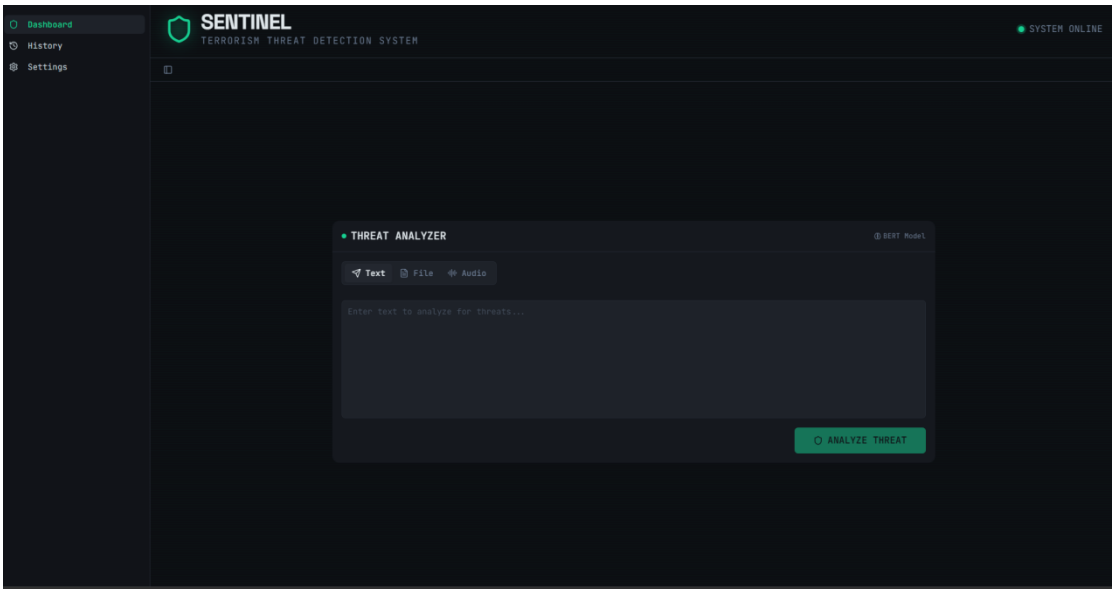

Figure 8 Threat Analyzer (Textual Format)

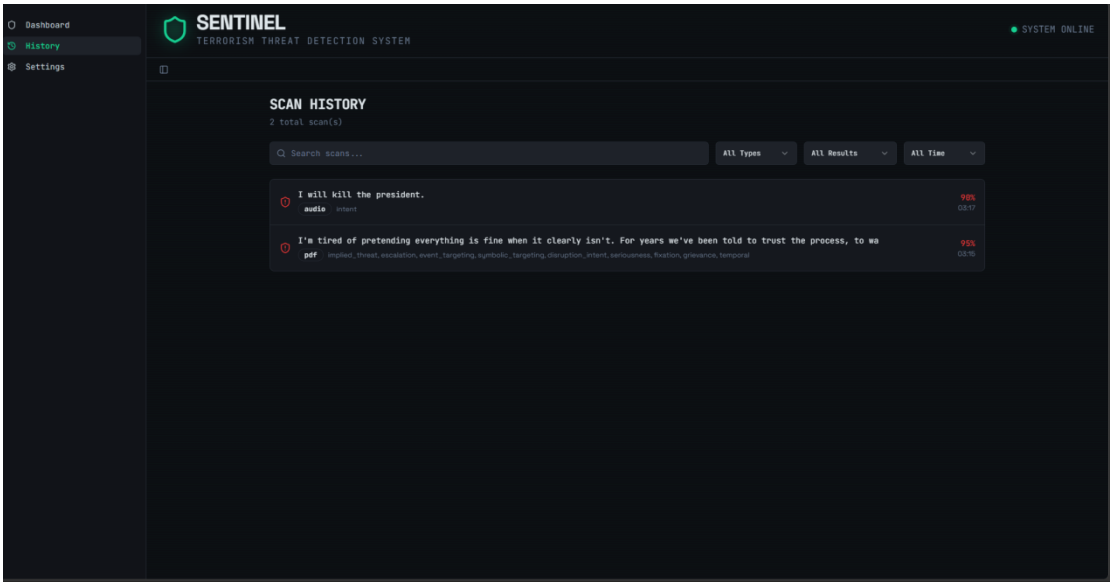

Figure 9 Scan History Page

7. Conclusion

This paper presented the design and implementation of a terrorism threat detection system leveraging Natural Language Processing, with BERT as the core classification model. The system offers a comprehensive and cohesive digital environment for the sophisticated detection and management of terrorism threats. From the initial high-security sign-in and registration portals to the multi-format Threat Analyzer dashboard, the system prioritizes a seamless user experience while maintaining a robust technical backbone. By integrating advanced ASR for audio transcription and specialized BERT-based models for deep linguistic analysis, the system transforms raw data into actionable security intelligence with high confidence. The inclusion of a detailed Scan History further ensures that all analyzed data is systematically archived, providing an essential audit trail for long-term threat monitoring and strategic risk mitigation.

The model achieved perfect evaluation scores across accuracy, precision, recall, F1 score, and MCC over three training epochs on the structured dataset, while user evaluation confirmed the system's reliability and response speed within the "Good" performance band. However, these results should be interpreted with caution given the constrained nature of the training dataset. Future work should focus on evaluating the model against larger, noisier, and more linguistically diverse datasets, incorporating multilingual support, and improving robustness to coded language, sarcasm, and sociocultural language variations.

Compliance with ethical standards

Acknowledgments

The authors acknowledge the support of the Department of Software Engineering, School of Computing, Babcock University, Nigeria, in the completion of this research.

Disclosure of conflict of interest

No conflict of interest to be disclosed.

Statement of ethical approval

This study was conducted in accordance with the ethical standards of the School of Computing, Babcock University, Nigeria. The research did not involve direct experimentation on human subjects. All datasets used for model training and evaluation including the Global Terrorism Database (GTD) and supplementary Kaggle datasets are publicly available and were accessed solely for academic research purposes. The user evaluation component involved four voluntary participants who were fully informed of the study's purpose and provided their consent prior to participation.

No personally identifiable information was collected, stored or disclosed at any stage of this research. The authors declare that all procedures comply with applicable institutional and national ethical guidelines for research integrity.

Statement of informed consent

Informed consent was obtained from all individual participants included in the user evaluation component of this study. Participants were briefed on the purpose of the research, the nature of their involvement, and their right to withdraw at any time without consequence. No personally identifiable information was collected during the evaluation process.

References

[1] Torregrosa J, Bello-Orgaz G, Martinez-Camara E, Del Ser J, Camacho D. A survey on extremism analysis using natural language processing: Definitions, literature review, trends and challenges. Journal of Ambient Intelligence and Humanized Computing. 2022. https://doi.org/10.1007/s12652-021-03658-z

[2] Guler A, Akgul I. Detecting terror threat elements using natural language processing. Journal of Scientific Reports-B. 2024;10:1-12.

[3] Florea M, Potlog C, Pollner P, Abel D, Garcia O, Bar S, Naqvi S, Asif W. Complex project to develop real tools for identifying and countering terrorism: Real-time early detection and alert system for online terrorist content based on natural language processing, social network analysis, artificial intelligence and complex event processing. [Unpublished report].

[4] Bonde L, Dembele S. High accuracy location information extraction from social network texts using natural language processing. International Journal on Natural Language Computing (IJNLC). 2023;12(4):1-12. DOI: 10.5121/ijnlc.2023.12401

[5] Atoum MS, Alarood AA, Alsolami E, Abubakar A, Al Hwaitat AK, Alsmadi I. Cybersecurity intelligence through textual data analysis: A framework using machine learning and terrorism datasets. Future Internet. 2025;17(4):182. https://doi.org/10.3390/fi17040182

[6] Abdalsalam M, Li C, Dahou A, Kryvinska N. Terrorism attack classification using machine learning: The effectiveness of using textual features extracted from GTD dataset. Computer Modeling in Engineering and Sciences. 2023;138(2):1431-1461. https://doi.org/10.32604/cmes.2023.029911